

ENABLEMENT OF ROUTE TABLE ENTRIES

FIELD

[0001] The invention pertains to the field of routing and, more particularly, to the enablement of route tables entries.

BACKGROUND

[0002] A network may be formed by connecting a plurality of nodes to a routing fabric that includes a plurality of routers. A source node may transmit a packet to a destination node via the routing fabric. The routers within the routing fabric route the packet from the source node toward the destination node based on information stored in a route table.

[0003] The route table includes a list of entries that identify where to direct a packet based on the packet's destination address. When the packet from the source node is received by one of the routers, it decodes the packet to identify its destination address. The router then looks up the entry in a route table corresponding to the identified destination address. That route table entry indicates an output port to which the received packet is directed.

SUMMARY

[0004] The following presents a simplified summary of the invention in order to provide a basic understanding of some aspects of the invention. This summary is not an extensive overview of the invention. It is intended to neither identify key or critical elements of the invention nor delineate the scope of the invention. Its sole purpose is to present some general concepts of the invention in a simplified form as a prelude to the more detailed description that is presented later.

[0005] In one embodiment, the invention encompasses a method for modifying the routing of packets in a system where packets are routed among a plurality of nodes in accordance with a route table and a route table enable mask. The route table enable mask has entries that characterize the enablement of corresponding entries in the route table. A packet including an operation code and an update mask is received from one of the nodes. The route table enable mask is read from a memory location. A new route table mask is generated by

performing a logical operation on the update mask and the route table enable mask in response to the operation code and then is stored in the memory location.

[0006] In another embodiment, the invention encompasses a method for updating a route table enable mask. A packet including an indication to modify the route table enable mask is received from one of the nodes in a network. An update mask is produced and a new route table enable mask is generated by performing the appropriate logical operation on the update mask and the route table enable mask.

[0007] In yet another embodiment, the invention encompasses a router of a routing fabric for routing packets among a plurality of nodes. The router comprises an input port, a plurality of output ports, a memory for storing a route table and a route table enable mask, and a processor. The processor produces an update mask and generates a new route table enable mask by performing an appropriate logical operation on the update mask and the route table enable mask in response to the determination of which logical operation to perform.

[0008] In yet another embodiment, the invention encompasses a symmetric multiprocessor system comprising a plurality of cells having at least one processor and a routing fabric. The routing fabric has a crossbar for communicating a packet from one cell to another cell. The crossbar has a plurality of input ports coupled to one of the plurality of cells for receiving a packet comprising an update mask from the cell, a plurality of output ports coupled to one of the plurality of cells, a storage device for storing a route table having a plurality of route table entries and a route table enable mask having a plurality of route table enable mask entries corresponding to route table entries, and a processor for generating a new route table enable mask by performing a logical operation on the update mask and the route table enable mask.

BRIEF DESCRIPTION OF THE DRAWINGS

[0009] For the purpose of illustrating the invention, there is shown in the drawings a form that is presently preferred; it being understood, however, that this invention is not limited to the precise arrangements and instrumentalities shown.

[0010] Figure 1 is a block diagram of a network according to an embodiment of the invention;

[0011] Figure 2 is a data flow diagram illustrating a method of routing according to an embodiment of the invention;

[0012] Figure 3 is a flow chart illustrating a method for processing a received packet according to an embodiment of the invention;

[0013] Figures 4A, 4B, 4C and 4D illustrate methods of generating new route table enable masks; and

[0014] Figure 5 is a block diagram of a symmetric multiprocessor system according to an embodiment of the invention.

DETAILED DESCRIPTION

[0015] Referring to the drawings, in which like reference numerals indicate like elements, there is shown in Figure 1 a network 100 according to an embodiment of the invention. The network 100 includes a plurality of nodes 102 connected to a routing fabric 104 that includes a plurality of routers 106.

[0016] Each router 106 includes at least one input port 116, at least one output port 114, a processor 110 and memory 112. When a packet from one node 102 (the source node) is received on one of a router's 106 input ports 116, the router 106 routes the packet toward another node 102 (the destination node) based on information stored in a route table and a route table enable mask.

[0017] If the destination node is connected to the router that received the packet, the router may route the packet directly to the destination node. For example, if node A 102a (source node) transmits a packet to router A 106a that has a destination address identifying node B 102b (destination node), the router A 106a will route the received packet directly to node B 102b via one of its output ports. If the destination node instead is node X 102x and is not directly connected to the router A 106a that received the packet, the router A 106a may

route the packet to router D 106d on another output port or to router B 106b on yet another output port on a path toward the destination node X 102x.

[0018] A router 106 determines which output port to use for transmitting a packet toward a destination node based on entries in a route table and a corresponding route table enable mask. The process by which a router 106 determines which output port to use is described with reference to the data flow diagram 200 in Figure 2.

[0019] The router includes a route table 206 and a route table enable mask 208. The entries 212 in the route table enable mask 208 characterize the enablement of corresponding entries 210 in the route table 206. In this exemplary embodiment, the route table enable mask entries 212 enable or disable routing according to their corresponding route table entries 210. A router may include one or more route tables and route table enable masks. In an embodiment, a router includes a route table for each of its input ports. In an embodiment, a router includes a route table enable mask that characterizes the enablement of entries in a plurality of route tables. In an embodiment, a route table enable mask entry 212 characterizes the enablement of and corresponds to a plurality of route table entries 210.

[0020] When a router 106 receives a packet, it decodes the header from the packet using a packet header decoder 202. The header of the packet includes a destination ID that identifies the desired destination of the packet. A destination ID decoder 204 decodes the packet header to identify the destination ID (i.e., identifier or address) of the packet.

[0021] Once the destination ID of the packet is identified, the router 106 maps the destination ID to one of the entries 210 in the route table 206 to determine where to and whether to route the received packet. The mapping of a destination ID to a route table entry 210 is illustrated by the arrows 214 in Figure 2.

[0022] Each route table entry 210 identifies an output port to which a packet should be routed. Each route table enable mask entry 212 characterizes the enablement of at least one route table entry 210.

[0023] If the mask entry 212 corresponding to the mapped route table entry 210 is a "1," the route table entry 210 is enabled and the packet is routed to the output port identified by the route table entry 210. If the mask entry corresponding to the mapped route table entry is a "0," the route table entry 210 is disabled and the packet is discarded or dropped.

[0024] The route table enable mask may be changed during operation of the network to affect the routing of packets through the network. For example, if a node 102 in the network 100 is to be taken offline, it may send an update packet to all routers instructing them to update their route table enable masks 208 to disable routing to the offline node. A node 102 may similarly send an update packet when it comes online to enable routing to that node.

[0025] A method of modifying a route table enable mask is described with reference to the flow chart 300 of a method for processing a received packet shown in Figure 3. Embodiments of the invention encompass methods where multiple steps of the method illustrated in Figure 3 are combined or are implemented out of the illustrated order. In an embodiment, the method of Figure 3 is implemented by the processor 110 executing a corresponding program stored in the memory 112. In an embodiment, the method of Figure 3 is implemented in by the router in hardware, software, firmware, or a combination thereof.

[0026] A packet from one of the nodes 102 in the network 100 is received by a router 106 in step 302. The destination address or ID of the packet is decoded from the packet header in step 304. The destination address identifies a unique node or router address and is used to determine whether the packet is destined for the router that received it or for another node or router. If the destination address is identified in step 306 as designating another node or router in step 306, the packet is further processed by the router and routed toward the destination node as described above with regard to Figures 1 and 2. In particular, in step 308, the destination address is mapped to one of the route table entries. The output port is identified based on the route table entry in step 310 and the packet is transmitted via the identified output port in step 312 if the route table entry is enabled by a corresponding route table enable mask entry.

[0027] If the destination address of the received packet is identified in step 306 as designating the router that received the packet, an operation code (opcode) is decoded from the packet in step 314. The opcode indicates a function to be performed by the router in response to the received packet. The indicated function may be to update the route table enable mask or to perform another function. If the router determines in step 316 that the function is not to update the route table enable mask, then that other function is performed in step 318.

[0028] If the function indicated by the opcode is determined in step 316 to be to modify the route table enable mask, an update mask is produced in step 320 and the route table enable mask is read from a location in memory 112 in step 322. A determination is made in step 324 of which logical operation to perform on the update mask and the route table enable mask. A new route table enable mask is generated in step 326 by performing the appropriate logical operation on the (old) route table enable mask and the update mask and the new mask is stored in the memory location in step 328.

[0029] In an embodiment, only one packet for modifying a route table enable mask is processed at a time and packets to modify the mask received from other routers are buffered until the update of the mask is complete for the currently processed packet. In an embodiment, steps 322, 324, 326, and 328 may only be performed for one received packet at a time.

[0030] Exemplary methods of generating new route table enable masks are described with reference to Figures 4A and 4B. A router includes a route table in Figure 4A where a destination address identifying nodes X, Y, and Z are mapped to route table entries identifying output ports A, B, and C, respectively. A value of “1” in a route table enable mask entry enables a corresponding route table entry and a value of “0” disables a corresponding route table entry.

[0031] A method of disabling routing is described with reference to Figure 4A. The bit in the update mask 402 corresponding to output port B is a “0” and all other bits are “1”s. A logical AND operation is performed on the update mask 402 and the route table enable mask 208 to generate the new route table enable mask 404 where the only modified entry corresponds to output port B. The AND operation disables routing to output port B by changing its corresponding mask entry to a “0” (or keeping it at “0” if previously set to “0”) and all other mask entries remain the same.

[0032] A method of enabling routing is described with reference to Figure 4B. The bit in the update mask 406 corresponding to output port A is a “1” and all other bits are “0”s. A logical OR operation is performed on the update mask 406 and the route table enable mask 208 to generate the new route table enable mask 408 where the only modified entry corresponds to output port A. The OR operation enables routing to output port A by changing its corresponding mask entry to a “1” (or keeping it at “1” if previously set to “1”) and all other mask entries remain the same.

[0033] In an embodiment of the invention, the route table enable mask is updated in response to a packet received from a cell without the cell knowing the values of the current route table enable mask. The logical operation may be configured to set (1) or reset (0) a particular route table enable mask entry without regard to the status of the other entries in the mask, without the cell reading the mask in advance of generating the packet, and without synchronization of other cells that may choose to update the same mask.

[0034] Although described above with regard to Figures 4A and 4B as modifying a single entry in a route table enable mask 208, embodiments of the invention encompass modifying a plurality of entries for enabling or disabling a plurality of route table entries. For example, the update mask 402 in Figure 4A may include a plurality of “0” bits to disable a plurality of entries in the route table 206.

[0035] Embodiments of the invention also encompass enabling and disabling one or more entries of a route table in a single operation. For example, a logical exclusive-OR operation of the route table enable mask and an update mask may enable one or more entries and disable one or more entries in the route table. A method of enabling and disabling routing is described with reference to Figure 4C. A logical exclusive-OR (XOR) operation is performed on the update mask 410 and the route table enable mask 208 to generate the new route table enable mask 408 where the route table enable mask entry corresponding to output port A is enabled and the route table enable mask entry corresponding to output port B is disabled in a single operation.

[0036] The term “logical operation” as used herein encompasses arithmetic operations such as summation, division, multiplication and division. The update mask may be produced such that the arithmetic operation results in the desired new route table enable mask. With reference to the embodiment of Figure 4D, if the logical operation is a summation (SUM) operation, the update mask 414 and the route table enable mask 208 are the operands of the summation and the new route table enable mask 416 is the sum of the operands. In this example, with the entries corresponding to output port A being the least significant bit in the summation, the sum of the route table enable mask 208 and update mask 414 generates a new route table enable mask 416 having an entry corresponding to output port B changed from enabled to disabled and an entry corresponding to output port C changed from disabled to enabled.

[0037] The destination ID may identify both a router and one of its route table enable masks and function as both the destination ID and the opcode. In other words, in an embodiment where a destination ID identifies a particular mask, the processor 110 interprets the corresponding packet as a request to modify that mask. When the destination ID identifies both a router and one of its route table enable masks, steps 306, 314, and 316 may be combined into a single step of identifying which mask is selected to be modified.

[0038] In an embodiment, a packet received on an input port can indicate to modify a route table enable mask corresponding to any input port on a particular router. In an embodiment, the processor 110 validates a packet that includes an indication to modify a route table enable mask. The processor 110 validates packets by restricting packets received on an input port so they are only permitted to modify a route table enable mask corresponding to that input port. In an embodiment, the processor 110 restricts modification of a route table enable mask entry to nodes having a source address that corresponds to the corresponding route table entry to be enabled or disabled.

[0039] In an embodiment, a packet received by the router 106 includes the update mask and step 320 of producing the update mask comprises decoding the update mask from the received packet. In another embodiment, the step 320 of producing the update mask comprises generating the update packet based on the source address and the route table enable mask. For example, in a system where a node may only enable or disable its own corresponding route table entries, the source address identifies the route table entry and mask entry to be modified. The update mask is then generated by the processor 110 in response to the current route table enable mask entry for that node by generating an update mask that will toggle its value between enable and disable values.

[0040] In an embodiment, the router determines which logical operation to perform based on the received packet. The packet may include an opcode, for example, indicating which logical operation to perform. For example, an opcode may include one bit to indicate which of an AND and an OR operation to perform. In another embodiment, the router may determine which logical operation to perform based on the update mask. For example, in the case described above with reference to Figures 4A and 4B, the processor may determine to perform a logical AND operation in response to update mask having only one "0" value and may determine to perform a logical OR operation in response to update mask having only one "1" value. In yet another embodiment, the router determines which logical

operation to perform based on the current value of a route table enable mask entry to be modified by selecting a logical operation to change its current value to an opposite value.

[0041] The method of updating the route table enable mask may be applied to a symmetric multiprocessor system (SMP) 500 as shown in Figure 5. The SMP 500 includes a plurality of cells 502 having at least one processor 504. The cells 502 are coupled to a routing fabric 506 via a network interface 508. The routing fabric 506 includes a plurality of crossbars 510 as its routers for communicating packets from one of the plurality of cells 502 to another of the plurality of cells 502.

[0042] The crossbars 510 include a plurality of input ports (IP) 512, a plurality of output ports (OP) 514, a storage device 516 and a processor 518. The input ports IP are coupled to cells 502 or to other crossbars 510 for receiving packets from the cells 502 or crossbars 510. The output ports OP 514 are coupled to cells 502 or to other crossbars 510 for transmitting packets to the cells 502 or crossbars 510.

[0043] One or more route tables and corresponding route table enable masks are stored in the memory 516 of each crossbar 510. The processor 518 uses the route tables and the route table enable masks in accordance with the description above for routing packets from among the cells 502 and crossbars 510 and for updating the route table enable masks. In an embodiment, a crossbar includes a route table for each of its input ports and a route table enable mask for each of its route tables. In an embodiment, a route table includes an entry for each cell and at least one destination address is mapped to a route table entry.

[0044] The invention is described above with a route table enable mask entry of “1” enabling a corresponding route table entry and a value of “0” disabling a corresponding route table entry. The invention also encompasses embodiments where a route table enable mask entry value of “0” enables and a value of “1” disables one or more corresponding route table entries. The invention encompasses embodiments where a route table enable mask entry characterizes the enablement of routing according to route table entries and is not limited to the use of single bit values to characterize enablement.

[0045] Although the invention is described above as using logical AND and OR operations, the scope of the invention encompasses other logical operations. For example, an appropriate update mask may be generated to perform an exclusive-OR operation for generating a new route table enable mask. The invention is not limited to a particular embodiment of a node or a network. A node generates a packet received by a router for

updating its route table enable mask. Embodiments of the invention encompass a node being a computer, a process, and a router. Embodiments of the invention encompass the network being the Internet and a multiprocessor system.

[0046] The foregoing describes the invention in terms of embodiments foreseen by the inventors for which an enabling description was available, although insubstantial modifications of the invention, not presently foreseen may nonetheless represent equivalents thereto.